Cloud Computing: Web-Scale Information Processing with MapReduce

Jimmy Lin
The iSchool
University of Maryland

Thursday, May 29, 2008
HCIL Symposium

“Web-Scale” Problems

- Emergence of data-intensive problems:
  - Problems involving the Web (e.g., crawling, searching)
  - “Post-genomics era” in life sciences
  - Data from accelerator experiments, remote sensors, …
  - Web 2.0 applications
  - High-quality animation
- How can we practically tackle these problems?

How much data?

- CERN's LHC will generate 15 PB a year (2008)
- NOAA has ~1 PB climate data (2007)
- Wayback machine has ~2 PB (2006)
- Google processes 20 PB a day (2008)
- "all words ever spoken by human beings" ~ 5 EB

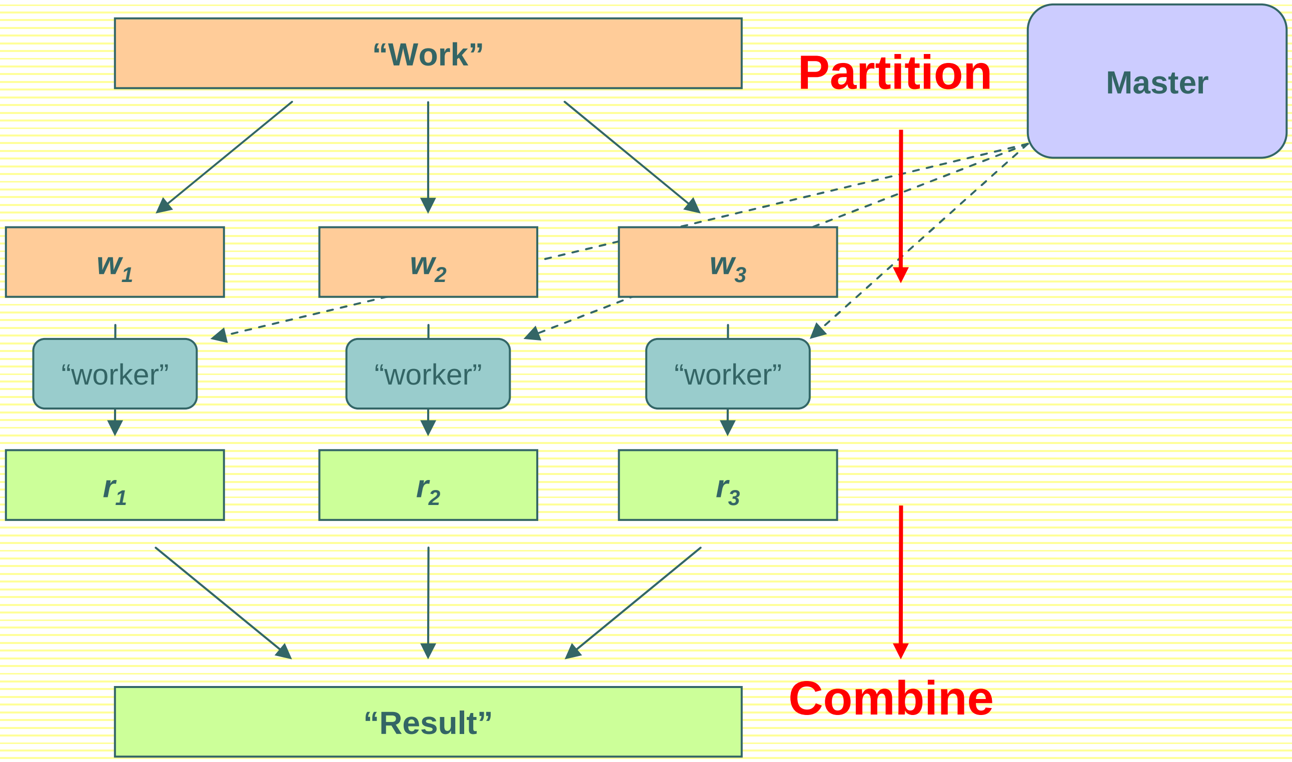
Relevance to HCI

- What does this have to do with HCI?
- Interfaces lies at the end of a long data analysis pipeline:
  - Collection, storage (by machines)
  - Aggregation, filtering, reduction (by machines)
  - Visualization, analysis, manipulation (by humans)

Humans can't interact with what machines can't crunch!

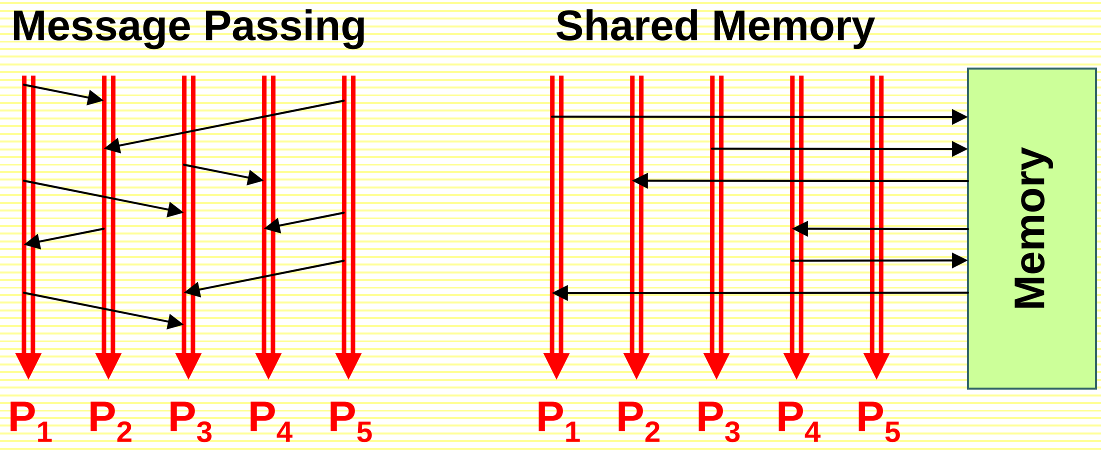
Only Practical Solution (Today)

- Divide and conquer
- Throw more machines at it



Challenges for Scaling Up

- Issues in parallel and distributed processing
  - Scheduling, data distribution, synchronization, …
  - Robustness, fault tolerance, …
- Traditional models for parallel and distributed programming are hard!



MapReduce in a Nutshell

- General problem structure:
  - Iterate over a large number of records
  - Extract something of interest from each
  - Shuffle and sort intermediate results
  - Aggregate intermediate results
  - Generate final output

Map

Reduce

- MapReduce provides a functional abstraction:
  - Programmer supplies “Mapper” and “Reducer”
  - Runtime automatically handles everything else!
- MapReduce deployments:
  - Google processes 20 PB a day with propriety implementation in C++
  - Hadoop is an open-source reimplementation in Java

MapReduce Runtime

- Handles scheduling
  - Assigns workers to map and reduce tasks
- Handles data distribution
  - Gets map workers to the data
- Handles synchronization
  - Shuffles data from mappers to reducers
- Handles faults
  - Detects worker failures and restarts
- Everything happens on top of distributed FS
  - GFS = Google File System

"Cloud Computing" Initiative

- Google/IBM's academic cloud computing initiative (October 2007)
  - Six initial pilot institutions: Washington, Berkeley, CMU, MIT, Stanford, UMD
- IBM provides UMD a Hadoop cluster
  - 20 machines (40 processors)
  - Couple of TB storage
  - Associated infrastructure support
- Maryland does good work with the cluster!
  - Use it to tackle open research problems
  - Use it in the classroom

Cloud Computing Course

- Attempt at integrating teaching and research
  - Pilot course in Spring 2008
  - Basic idea: Ph.D. students leading teams of masters and undergraduate students
  - Goal: tackle “Web-scale” research problems
  - Evaluation: generate publishable results
- What do students get out of it?
  - Everyone learns about MapReduce
  - Undergraduates learn about a particular research area
  - Ph.D. students gain mentoring experience (and help with their work)

Ongoing Projects

- Six teams:
  - Statistical machine translation
  - Reference resolution in email archives
  - Language modeling
  - Biomedical text retrieval
  - Text-image separation in children's books
  - Biological sequence alignment
- Thirteen students
  - 3 undergrads, 3 Masters, 7 Ph.D.
  - From the iSchool, CS, Linguistics, Geography
- Many campus labs involved
  - NLP/IR, HCI, computational biology, etc.

Statistical Machine Translation

Chris Dyer (Ph.D. student, Linguistics)
Aaron Cordova (undergraduate, Computer Science)
Alex Mont (undergraduate, Computer Science)

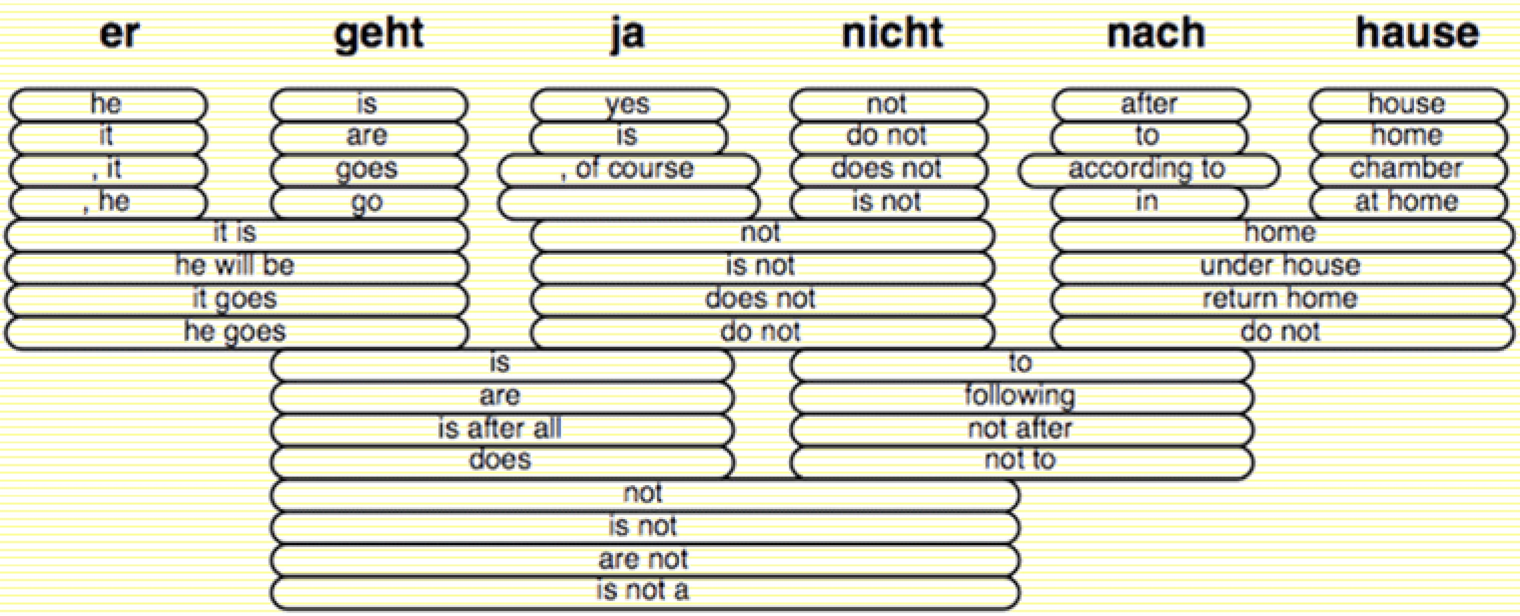
- Conceptually simple:
  (translation from foreign *f* into English *e*)

$$\hat{e} = \arg\max_{e} P(e \mid f)$$

$$\hat{e} = \arg\max_{e} P(f \mid e) P(e)$$

- Difficult in practice!
- Phrase-Based Machine Translation (PBMT) :
  - Break up source sentence into little pieces (phrases)
  - Translate each phrase individually

Phrasal Decomposition



Example from Callison-Burch (2007)

MT Architecture

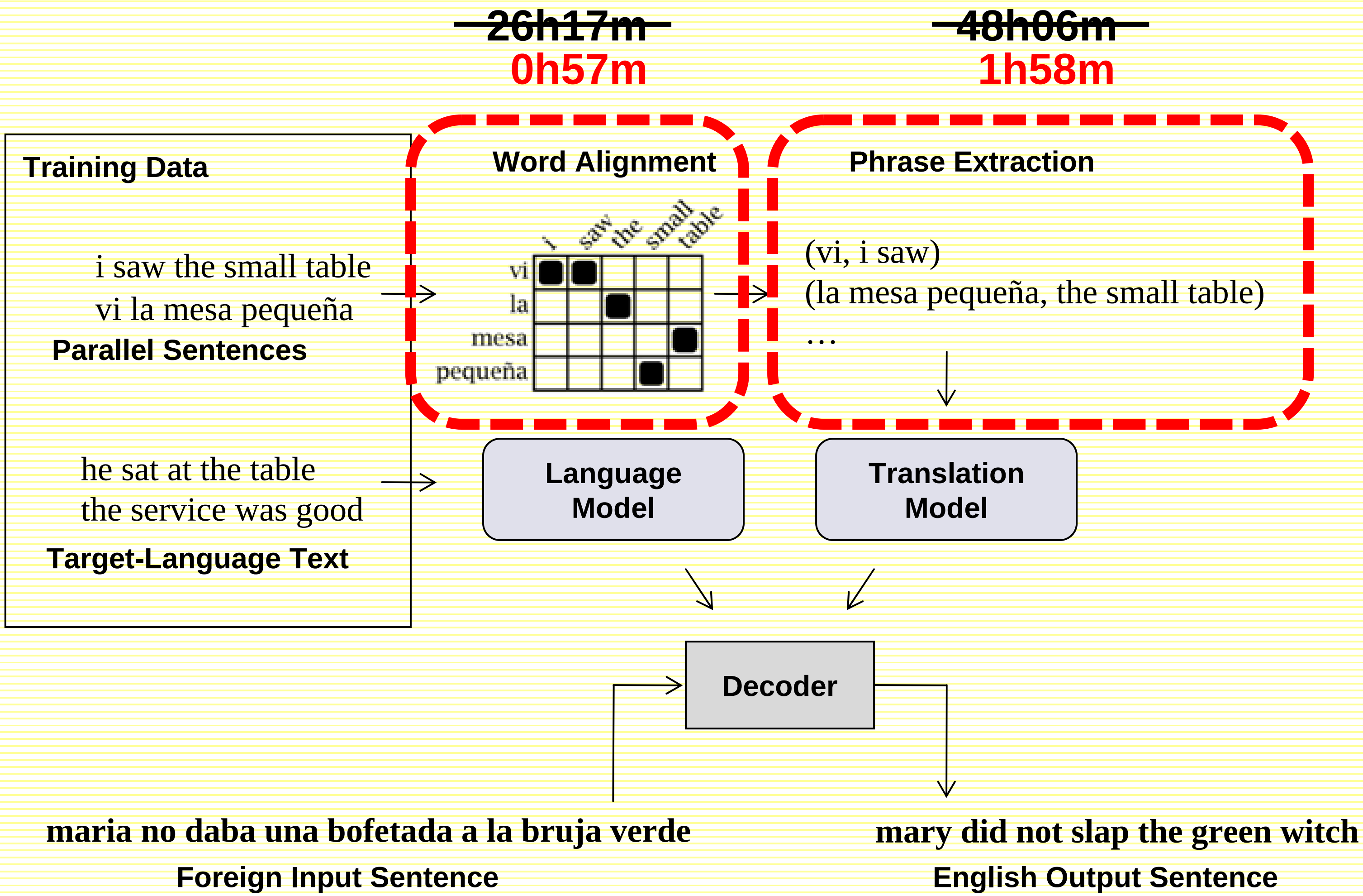
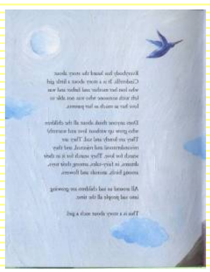


Image Processing

Chang Hu (Ph.D. student, Computer Science),
Punit Mehta (undergraduate, Computer Science)

- Text-image separation in children's books
  - International Children's Digital Library (ICDL): 2412 books, 155,962 pages (and counting)



Hand transcription to improve readability

- Successful implementation in Hadoop
  - Takes advantage of distributed file system

Want to learn more?

- Come to my tutorial tomorrow!
- Topics covered:
  - Overview of parallel and distributed computing
  - Functional programming
  - MapReduce and the Google File System
  - Graph Algorithms with MapReduce
  - Information Retrieval Algorithms with MapReduce

Acknowledgements

- **Google:** Christophe Bisciglia, et al.
- **IBM:** Dennis Quan, Eugene Hung, et al.
- Thirteen bright students from UMD:

Chris Dyer
(Linguistics Ph.D.)

Alex Mont
(CS ugrad)

Aaron Cordova
(CS ugrad)

Chang Hu
(CS Ph.D.)

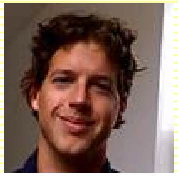
Punit Mehta
(iSchool masters)

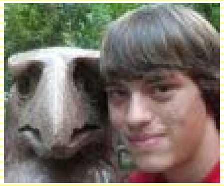
George Caragea
(CS Ph.D.)

Christiam Camacho
(iSchool masters)

Tamer Elsayed
(CS Ph.D.)

Greg Jablonski
(iSchool masters)





Alan Jackoway
(CS ugrad)

Denis Filimonov
(Linguistics Ph.D.)

Hua Wei
(Geography Ph.D.)

Mike Schatz
(CS Ph.D.)